An Intention-Based Language for Representing Clinical Guidelines

Yuval Shahar, M.D., Ph.D., Silvia Miksch, Ph.D., and Peter Johnson, M.D.
Section on Medical Informatics and Knowledge Systems Laboratory,
Stanford University, Stanford, CA

Automated support for guideline-based care would be enhanced considerably by a standard representation of clinical guidelines. To faciliate use and reuse, we suggest a representation that includes the explicit intentions *of the guideline's author. These intentions include the desirable actions of the care provider and the patient states to be achieved before, during, and after the administration of the guideline.* Intentions *are temporal patterns of provider actions or patient states to be maintained, achieved, or avoided. We view automated support as a collaborative effort of the health-care provider and an automated assistant and involves several different tasks. We defined the syntax and the semantics of a text-based language* (ASBRU) *for representation and annotation of clinical guidelines. The language supports maintenance of the automated assistant's knowledge base and could improve the quality and flexibility of the automated assistant's recommendations. In the* ASGAARD *project, we are developing reasoning mechanisms that use the ASBRU language for execution and critiquing tasks in conjunction with online electronic patient medical records.*

INTRODUCTION: AUTOMATED SUPPORT OF GUIDELINE-BASED CARE

There have been several efforts to provide automated support for protocol-based care and, in general, for guideline-based care. In the prescriptive style, care providers are given active interpretation of the guideline [1–3]; in the critiquing style, the program comments on the physician's plan, rather than recommending one of its own [4,5]. Other approaches permit hypertext browsing of guidelines via the World Wide Web, or use a rule-based system to provide situation-specific recommendations based on the user's manual entry of clinical information into computer-generated forms [6,7], but do not use the patient's electronic medical record. Some approaches encode guidelines as elementary state-transition tables or as situation–action rules dependent on the electronic medical record [8], but do not include an intuitive representation of the guideline's clinical logic, and have no semantics for the different types of clinical knowledge represented. None of these systems have a representation of guidelines that (1) has knowledge roles specific to the guideline-based care task, (2) is machine and human readable, and (3) allows data stored in an electronic patient record to invoke an application that executes the guideline's logic and related tasks, such as critiquing. A standard, human- and machine-readable representation of clinical guidelines, that has an expressive syntax and task-specific semantics, combined with the ability to interpret that representation in automated fashion, would facilitate guideline dissemination, (online) accessibility, and (automated) applicability. Such a representation also would support additional reasoning tasks, such as critiquing, quality assurance [9], and evaluation. A standard representation specific to the task of guideline-based care would also facilitate authoring and modification.

Automated support for the application of a clinical guideline involves a dialog between a care provider and an automated assistant. Each has relative advantages. The provider has access to additional data apart from the electronic medical record and has broader knowledge. The automated assistant might more easily detect patterns in the data over long periods of time, and has fast and direct access to the guideline's complex rules and procedures. The aim is *synergy*, exploiting the best of both participants. Such a collaboration would benefit from an explicit representation of the *intentions* of the guideline's author to enable, for example, execution-time modifications that preserve the spirit of the guideline. In addition, automated reasoning mechanisms need several types of domain-specific knowledge, such as effects of interventions (e.g., drug administrations) and legitimate modifications to guidelines.

Guidelines are often ambiguous or incomplete. A diabetes guideline might recommend a therapy target without any specific recommendations on ways to achieve it, or might suggest a drug without a precise dose. A chemotherapy protocol might specify what to do when renal toxicity *or* when suppression of the bone marrow occur, but not what to do when *both* occur. Physicians sometimes do not adhere to protocols, believing their actions to be closer to intentions of protocol designers [10]. The automated assistant should recognize cases in which the provider's actions still adhere to the overall intentions and policies, and should adjust accordingly its critique or other support. Increased flexibility would enable the automated assistant to continue offering useful advice even if the default guideline is not followed literally (e.g., there is a reasonable alternative to the prescribed therapy).

A useful standard guideline-execution language needs to be expressive with respect to representations of time-oriented, possibly continuous actions and needs to have a rich set of parallel and sequential operators. The language requires well-defined semantics for both the prescribed actions and the intentions and preferences underlying them. Thus, the provider's actions can be better supported, leading to a more flexible dialog and to a better acceptance of automated systems for guideline-based care support.

A DESIGN-TIME VERSUS EXECUTION-TIME CRITIQUING MODEL

During **design time** of a guideline, an **author** (or a committee) designs a guideline (Figure 1). The author prescribes actions (e.g., administer terbutaline in the morning and in the evening), an intended plan (the overall temporal pattern of actions; e.g., use a β_2 agonist twice a day), and the intended patient states (e.g., peak respiratory flow should be at least 70% of predicted). **Intentions** are goals at various levels of the guideline: temporal patterns of provider actions or patient states, to be achieved, maintained, or avoided.

During **execution time**, a **care provider** performs actions, which are recorded, observed, and abstracted over time into an abstracted plan (see Figure 1). The state of the patient also is recorded, observed, and abstracted over time. Finally, the intentions of the care giver might be recorded too—inferred from her actions or explicitly stated.

The five comparisons shown in Figure 1 define different **behaviors** of the execution. A care provider might not follow the precise *actions*, but still follow the intended *plan* and achieve the desired states. A provider might even not follow the overall plan, but still adhere to a higher-level *intention*. Alternatively, the provider might be executing the guideline correctly, but the patient's state might differ from the intended, perhaps indicating a complication that needs attention or a failure of the guideline.

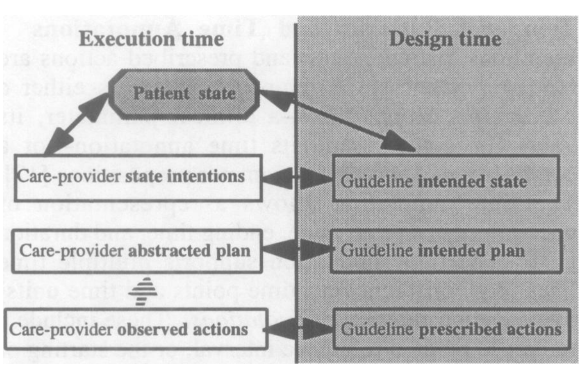

Figure 1. Design time versus execution time.

THE GUIDELINE-BASED CARE SUPPORT TASKS

Based on an intention-based critiquing model, we can describe some of the tasks relevant to the support of guideline-based execution, and analyze their knowledge requirements (Table 1). The (syntactic) verification and (semantic) validation tasks are performed at design time. The rest of the tasks are performed at execution time. Tasks can be formulated as answering a specific set of questions (see Table 1).

The semantics of the different types of intentions, preferences, and conditions used in our execution language are discussed in the next section. Given these annotations, we can define the types of knowledge required to solve each task (see Table 1).

The following example demonstrates the tasks of plan-recognition and critiquing. During therapy of an oncology patient, severe anemia is detected for the second consecutive week, the protocol's prescribed action is to attenuate the dose of a drug toxic to the bone marrow, and the provider gives the patient a transfusion of blood. This action seems to contradict the prescribed action. However, the automated assistant can note that the transfusion *increases* the value of the hemoglobin level directly through an external intervention, while the protocol's recommendation *increases* the value of the same parameter by *reducing* the magnitude of an intervention (i.e., the dose of the toxic drug) that *decreases* its value. The assistant also notes that the state intention was "avoid a severe anemia period of more than 2 weeks duration." Therefore, the provider is still following the intention of the protocol. By recognizing this high-level intention and its achievement by a different strategy, the automated assistant can accept the provider's alternate set of actions, and even support these actions. Such an ability can increase the usefulness of guideline-based decision-support systems to clinical practitioners, and assist in modifying guidelines when execution of the recommended actions is impossible or underspecified. Note that we assume knowledge about the *effects of interventions* on clinical parameters, and knowledge of domain-independent and domain-specific *guideline-revision strategies*. Both intervention effects and revision strategies can be represented formally [11].

A subtask implicit in several of the tasks in Table 1 is the abstraction of higher-level clinical concepts from time-stamped patient data during the execution of the guideline. We have defined a formal, domain-independent framework for solving this task, the *knowledge-based temporal-abstraction method*. The method has been implemeted as the **RÉSUMÉ** system and evaluated in several clinical domains [12].

Table 1: Several guideline-support tasks and the knowledge required to solve them.

Task	Questions to be answered	Required Knowledge
Verification	Are the intended plans chievable by following the prescribed actions? (*a syntactic check*)	Prescribed actions; intended overall action pattern (i.e., the plan)
Validation	Are thė intended states achievable by the prescribed actions and intended plan? (*a semantic check*)	Prescribed actions, intended overall action pattern; intended states; action/plan effects
Applicability of guidelines	What guidelines or protocols are applicable this time to this patient?	Filter and setup preconditions; overall intended states; the patient's state
Execution of guideline	What should be done now according to the guideline's prescribed actions?	Prescribed actions and their filter and setup preconditions; suspension, restart, completion, and abort conditions; the patient's state
Recognition of intentions	Why is the care provider executing a particular set of actions, especially if those deviate from the guideline's prescribed actions?	Executed actions and their abstraction to executed plans; action and state intentions; the patient's state; action/plan effects; revision strategies; preferences
Critique of the provider's actions	Is the care provider deviating from the prescribed actions or intended plan? Are the deviating actions compatible with the author's plan and state intentions?	Executed actions and their abstraction to plans; action and state intentions of the original plan; the patient's state; action/plan effects; revision strategies; preferences
Evaluation of guideline	Is the guideline working?	Intermediate/overall state intentions; the patient's state; intermediate/overall action intentions; executed actions and plans
Modification of an executing guideline	What alternative plans are relevant at this time for achieving a given state intention?	Intermediate/overall state intentions; action/plan effects; filter and setup preconditions; revision strategies; preferences; the patient's state

In the **ASGAARD** project, we are developing different reasoning modules that solve the guideline-based care tasks shown in Table 1. The task-specific reasoning modules require different types of knowledge. For instance, the knowledge-based temporal-abstraction method implemented by the RÉSUMÉ module requires knowledge about temporal-abstraction properties of clinical parameters, such as persistence of their values over time [12].

The specifications of clinical guidelines and of their independent components (we refer to either of these entities as **plans** in this paper) are all represented uniformly and organized in a *guideline-specification library*. Figure 2 presents the overall architecture.

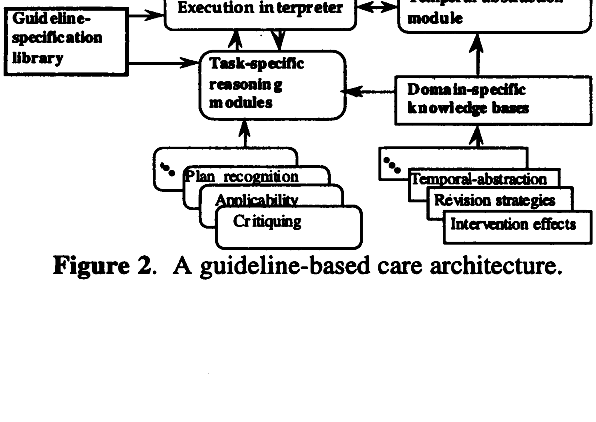

Figure 2. A guideline-based care architecture.

ASBRU: AN INTENTION-BASED EXECUTION LANGUAGE

We developed a guideline-execution language, called **ASBRU**. In ASBRU, prescribed actions can be continuous, guideline plans might be executed in parallel and/or in sequence, temporal scopes and parameters of clinical interventions can be flexible, and explicit intentions and preferences can underlie the guideline. These features are in contrast to many traditional plan-execution representations, which assume instantaneous actions and effects. However, clinical interventions often are continuous and might have delayed effects. We have defined a formal syntax for the language in standard Backus-Naur form (BNF).

Temporal Patterns and Time Annotations

Intentions, patient states, and prescribed actions are temporal patterns. A temporal pattern is either a *parameter proposition*—a clinical parameter, its value, its context, and its time annotation—or a combination of multiple parameter propositions [12]. The time annotation allows a representation of uncertainty in starting time, ending time, and duration [13]. The time annotation supports multiple time lines (e.g., different zero-time points and time units) by providing *reference annotations*. These include a reference point, a reference interval, or the starting or the finishing of a previous executed plan (e.g., start plan A 20 minutes after having finished plan B). Time stamps can be relative (e.g., 20 minutes after an action), can be absolute (e.g., 8:00 am, January 17, 1996), can include cyclic abstractions of absolute time (e.g., MORNINGS), can be domain-dependent (e.g., DELIVERY), and can use domain-dependent units (e.g., GESTATIONAL-WEEKS). We allow short-cuts such as when a plan should start immediately or when a condition should hold during the whole plan. This notation enables the expression of interval-based intentions, patient states, and prescribed actions with uncertainty regarding starting, finishing, or length, and time intervals using different granularity and reference points.

The Syntax and Semantics of ASBRU

A guideline plan is composed of a set of plans. A semantic stop condition is applied to terminate the recursion: the recursion stops if no decomposition of the plan is found in the plan library, thus representing a nondecomposable plan (also called an *action*). Such a plan is referred to the provider for execution. During the execution phase, an applicable plan is instantiated. At execution time, a set of mutually exclusive **plan states** describes the status of a plan instance. **State-transition criteria** specify transitions between states. The set of states is {*started, completed, suspended, restarted, aborted*}.

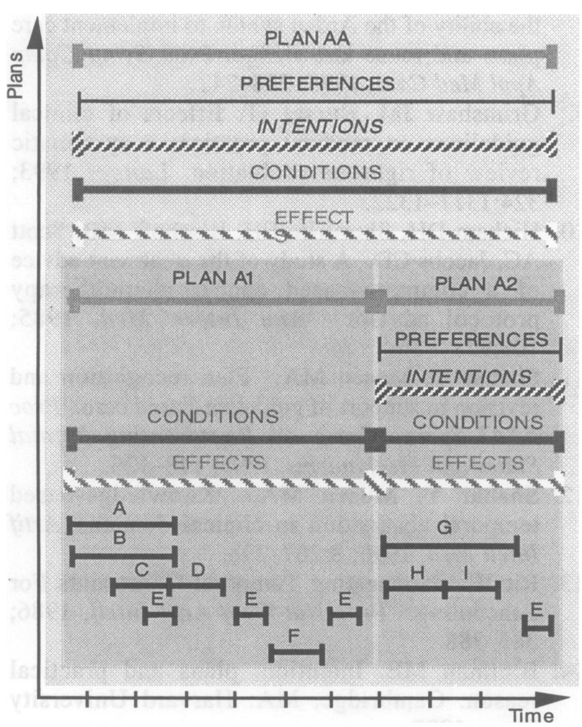

Figure 3. Graphical representation of a clinical-guideline specification represented in the intention-based language. Plan AA includes plans A1, A2 in sequence; plan A1 includes (in parallel) plans A, B, and C, and repeated action E.

A plan consists of a name and five components: **preferences, intentions, conditions, effects,** and a **plan body** which describes the actions to be executed (Figure 3). All components are optional.

The **plan body** can be sequential (a linear set of plans), parallel (a set of plans performed together or in any order), or cyclical. Each subplan has the same structure (see Figure 3). Figure 4 shows a portion of the representation of a guideline used at the Stanford University Medical Center for management of non-insulin-dependent gestational diabetes mellitus (GDM). The plan body consists of three plans whose execution starts together. These plans are decomposable into other plans, which are available in the plan library.

Preferences bias or constrain the selection of a plan (e.g., utility function, resource restriction).

```
(PLAN observing-NID-GDM
(DOMAIN-DEPENDENT TIME-ASSIGNMENT
  (TIME-SHIFTS DELIVERY <- 38 WEEKS)
  (TIME-POINT CONCEPTION
      <- (ask (ARG "conception-date?")))
(PREFERENCES
    (SELECT-METHOD EXACT-FIT))
(INTENTION:INTERMEDIATE-STATE
  (MAINTAIN STATE(blood-glucose)
    (NORMAL|SLIGHTLY-HIGH) GDM-Type-II
    [[24 WEEKS, 24 WEEKS],
    [DELIVERY, DELIVERY],[_,_],
    CONCEPTION]))
(INTENTION:OVERALL-STATE
  (AVOIDED STATE(blood-glucose) HIGH
    GDM-Type-II [[24 WEEKS, 24 WEEKS],
    [DELIVERY, DELIVERY], [7 DAYS,_],
    CONCEPTION]))
(SETUP-PRECONDITIONS
  (PLAN-STATE one-hour-GTT COMPLETED
    [[24 WEEKS, 24 WEEKS],
    [26 WEEKS, 26 WEEKS],[_,_],
    CONCEPTION]))
(FILTER-PRECONDITIONS
  (one-hour-GTT (140, 200) pregnancy
    [24 WEEKS, 24 WEEKS],
    [26 WEEKS, 26 WEEKS],[_,_],
    CONCEPTION]))
(SUSPEND-CONDITIONS (OR STARTED RESTARTED)
  (STATE(blood-glucose) HIGH GDM-Type-II
    [[24 WEEKS,24 WEEKS],
    [DELIVERY, DELIVERY], [4 DAYS,_],
    CONCEPTION]
    (SAMPLING-FREQUENCY 30 MINUTES)))
(ABORT-CONDITIONS
  (OR STARTED SUSPENDED RESTARTED)
    (insulin-indicator-conditions TRUE * *
    (SAMPLING-FREQUENCY 30 MINUTES)))
(COMPLETE-CONDITIONS (OR STARTED RESTARTED)
  (delivery TRUE GDM-Type-II *
    (SAMPLING-FREQUENCY 30 MINUTES)))
(DO-ALL-TOGETHER
  (glucose-monitoring)
  (nutrition-management)
  (observe-insulin-indicators)))
```

Figure 4. A portion of the representation of a guideline for management of non-insulin-dependent gestational diabetes mellitus (GDM).

Intentions include four categories of temporal patterns of care-provider actions or patient states to be maintained, achieved, or avoided: **intermediate state** (e.g., maintain the blood-glucose state within the range [normal, slightly high] until delivery [Figure 4]), **overall state pattern** (e.g., avoid more than 7 days of high blood-glucose state [Figure 4]), **intermediate action** (e.g., monitor blood glucose four times a week), and **overall action pattern** (e.g., visit the dietitian regularly).

Conditions are temporal patterns that define the transition conditions between neighboring plan states (e.g., *suspend* conditions determine when a started plan has to be suspended). In addition, **setup-preconditions** have to be achieved to enable starting a plan (e.g., patent has an intravenous line), and **filter-preconditions** have to hold initially if the plan is applicable (e.g., patient is pregnant).

Effects describe the effects of an action or an overall plan on clinical parameters (e.g., the dose of insulin is inversely related to blood glucose).

SUMMARY AND DISCUSSION

Intentions have long been a subject of interest for researchers in philosophy and computer science [14] We are building on such work and extending it.

Representing clinical guidelines and their underlying intentions in a standard, machine-readable, and machine-interpretable way is crucial for dissemination of clinical knowledge, and can improve the quality of care [9]. The representation we suggest (ASBRU) supports several *knowledge roles* that can be used by multiple reasoning modules, both for direct execution of a guideline and for related tasks, such as recognition of physician's intentions and for critiquing her plans. ASBRU also is very expressive regarding time-oriented actions and patient states.

Physicians need not have familiarity with ASBRU to author guidelines. Graphic knowledge-acquisition tools can be generated automatically by systems such as PROTÉGÉ-II [15], given a specification of the language. We are developing such a tool. In the ASGAARD project, we are focusing on the development of the execution and critiquing modules.

Acknowledgments

This work has been supported by grants LM05305, LM05708, and LM06245 from the NLM, IRI-9528444 from the NSF, ARPA grant N66001-94-D-6055, and J01042-MAT from the FWF Erwin Schrödinger Auslandstipendium.

References

1. Musen MA, Carlson CW, Fagan LM, Deresinski SC, Shortliffe EH. T-HELPER: Automated support for community-based clinical research. *Proc Symp Comp Appl Med Care*, 1992;719-723.
2. Tu SW, Kahn MG, Musen MA, Ferguson, JC, Shortliffe EH., Fagan LM. Episodic skeletal-plan refinement based on temporal data. *Comm ACM*, 1989; 32:1439–1455.
3. Herbert SI, Gordon CJ, Jackson-Smale A, Renaud-Salis J-L. Protocols for clinical care. *Comp Meth Prog Bio*, 1995; 48:21–26.
4. Miller PL. Expert critiquing systems: practice-based medical consultation by computer. New York, NY: Springer-Verlag, 1986.
5. Van der Lei J, Musen MA. A model for critiquing based on automated medical records. *Comput Biomed Res*, 1991; 24:344–378.
6. Barnes M, Barnett GO. An architecture for a distributed guideline server. *Proc Symp Comp Appl Med Care*, 1995; 233–237.
7. Liem EB, Obeid, JS, Shareck, PE, Sato L, Greenes RA. Representation of clinical practice guidelines through an interactive world-wide-web interface. *Proc Symp Comp Appl Med Care,* 1995; 223–227.
8. Sherman EH, Hripcsak G, Starren J, Jenders RA, Clayton P. Using intermediate states to improve the ability of the Arden syntax to implement care plans and reuse knowledge. *Proc Symp Comp Appl Med Care*, 1995; 238–242.
9. Grimshaw JM, Russel IT. Effects of clinical guidelines an medical practice: a systematic review of rigorous evaluation. *Lancet* 1993; 324:1317–1322.
10. Hickam DH, Shortliffe EH, Bischoff MB, Scott AC, Jacobs CD. A study of the treatment advice of a computer-based cancer chemotherapy protocol advisor. *Ann Intern Med*, 1985; 103:928–936.
11. Shahar Y, Musen MA. Plan recognition and revision in support of guideline-based care. *Proc AAAI Spring Symp on Representing Mental States and Mechanisms*, 1995; 118–126.
12. Shahar Y, Musen MA. Knowledge-based temporal abstraction in clinical domains, *Artif Intell Med*, 1996; 8:267–298.
13. Rit JF. Propagating Temporal Constraints For Scheduling. *Proc Nat Conf Artif Intell*, 1986; 383–388.
14. Bratman ME. Intention, plans and practical reason. Cambridge, MA: Harvard University Press, 1987.
15. Musen MA, Gennari JH, Eriksson H, Tu, SW, Puerta AR. PROTÉGÉ-II: Computer support For development of intelligent systems from libraries of components. *Proc World Cong Med Inform*, 1995; 766–770.